Taxon

**Taxon (Taxonomical Unit):** is an entity whose similarity (or dissimilarity) can be numerically measured. E.g., Species, Populations, Genera, Amino Acid Sequences, Nucleotide Sequences, Languages.

Phylogeny is an organization of the taxa in a rooted tree, with distances assigned to the edges in such a manner that the “tree-distance” between a pair of taxa equals the numerical value measuring their dissimilarity. The dissimilarity and the edge lengths of the phylogenetic trees can be related to the rate of evolution (perhaps determined by a molecular clock).

**Comparing a Pair of Taxa**

**Discrete Characters:** Each taxon possesses a collection of characters and each character can be in one of finite number of states. One can describe an $n$ taxa with $m$ characters by an $n \times m$ matrix over the state space. **Character State Matrix.**

**Comparative Numerical Data:** A distance is assigned between every pair of taxa. One can describe the distances between $n$ taxa by an $n \times n$ matrix over $\mathbb{R}_+$. **Distance Matrix.**

**Character States**

**Some Assumptions:**
- The characters are inherited independently from one another.
- Observed states of a character have evolved from one “original state” of the nearest common ancestor of a taxon.
- Convergence or parallel evolution are rare. That is the same state of a character rarely evolve in two independent manners.
- Reversal of a character to an ancestral state is rare.

**Classifying Characters**

**Characters:**
- **Unordered / Qualitative Character:** All state transitions are possible.
- **Ordered / Cladistic Character:** Specific rules regarding state transition are assumed.

  *Linear Ordering*

  *Partial Ordering* (with a derivation tree).
Perfect Phylogeny

A phylogenetic tree $T$ (with edges labeled by state transitions) is called perfect, if it does not allow reversal or convergence—that is, with respect to any character $c$, and any pair of states $w$ and $s$ at most one edge is labeled $w \rightarrow s$ or $s \rightarrow w$.

**Example:** Binary characters with two states (0=ancestral, and 1=derived): any character $c_i$ labels at most one edge and implies a transition from 0 $\rightarrow$ 1 in the $i$th position.

**Perfect Phylogeny Problem:**
- **Given:** A set $O$ with $n$ taxa, a set $C$ of $m$ characters, each character having at most $r$ states.
- **Decide:** If $O$ admits a perfect phylogeny.

A set of defining characters are compatible, if a set of objects defined by a character set matrix admits a perfect phylogeny.

Compatibility Criteria

Allow reversal and convergence properties in the models of evolution.

**Parsimony Criteria:** Minimize the occurrences of reversal and convergence events in the reconstructed phylogeny tree.

- **Dollo Parsimony Criterion:** Minimize reversal while forbidding convergence.
- **Camin-Sokal Parsimony Criterion:** Minimize convergence while forbidding reversal.

**Compatibility Criteria:** Exclude minimal number of characters under consideration so that the reconstructed phylogeny tree is perfect and does not admit any occurrence of reversal or convergence.

Computational Infeasibility

**Perfect Phylogeny Problem** for arbitrary ($>2$) number of unordered characters and arbitrary ($>2$) number of states is NP-complete.

**Optimal Phylogeny Problem under compatibility criteria** is NP-complete.

**Optimal Phylogeny Problem either under Dollo or Camin-Sokal parsimony criteria** is NP-complete.

Binary Character Set

Each character has two states $\{0, 1\}$

If a character is ordered then $0 \rightarrow 1$ (0=ancestral and 1=derived), or converse.

For binary characters (ordered or unordered), perfect phylogeny problem can be solved efficiently

- **Poly time, for $n$ taxa and $m$ characters, Time $= O(nm)$.

A two phase algorithm:

- **Perfect Phylogeny Decision Problem**
- **Perfect Phylogeny Reconstruction Problem**

Compatibility Condition
\( T = \text{Perfect Phylogeny for } M \text{ iff } \)
\[
\begin{align*}
(8 \ c_i = \text{character}) & (9! \ e = \text{tree-edge}) \ \text{label}(e) = \{c_i, 0! 1\} \\
\text{root}(T) = (0, 0, 0, \ldots, 0)
\end{align*}
\]

A path from root to a taxon \( t \) is labeled \((c_{i_1}, c_{i_2}, \ldots, c_{i_j})\)

\( t \) has 1's in positions \( i_1, i_2, \ldots, i_j \).

**Perfect Phylogeny Condition**

\( M = n \times m \) Character State Matrix, \( j \in \{1..m\} \)

\( O_j = \{ i = \text{taxon} : M_{ij} = 1 \} \)

\( O^c_j = \{ i = \text{taxon} : M_{ij} = 0 \} \)

**Key Lemma**

**Lemma:** A binary matrix \( M \) admits a perfect phylogeny iff

\[
(8 i, j \in \{1, m\}) \ O_i \cap O_j = \emptyset \text{ or } O_i \cup O_j \text{ or } O_i \mathbin{\#} O_j
\]

**Proof:** \( (\text{By induction, Base case } m=1 \text{ is trivial. Induction case, } m=k+1:}) \)

\( T_k = \text{Tree for } k \text{ characters. } O_{k+1} \) is contained in a subtree with minimal # taxa rooted at \( r \). \( r \) must be a leaf node. Either an edge needs to be labeled or the subtree rooted at \( r \) has to be split.

**Simple Algorithm based on the Lemma**

Compare every pair of columns for the intersection and inclusion properties. Total of \( O(m^2) \) pairs, each comparison can be done in \( O(n) \) time.

Total Time Complexity = \( O(nm^2) \)

Can be improved to \( O(nm) \) time.

**Improved Decision Algorithm**

**Algorithm**

First radix sort columns of \( M \) based on the number of 1’s in each column.

for each \( L_{ij} \) do \( L_{ij} := 0; \)

for \( i := 1 \) to \( n \) do

\( k := -1; \)

for \( j := 1 \) to \( m \) do

if \( M_{ij} = 1 \) then \( L_{ij} := k; k := j \)

for each column of \( j \) of \( L \) do

if \( 9 \ i \mid L_{ij} \) \( L_{ij} \) and both nonzero then return False

return True.

**Example**

**Reconstruction Algorithm**

**Two Characters**
An $n \leq 2$ Character State Matrix with arbitrary number of states admits a perfect phylogeny iff its corresponding state intersection graph (SIG) is acyclic. 

State Intersection Graph: For each state $s$ of character $c_i$ create a vertex $v$ of $G$. Let $O_v = \{ t : M_{ij} = s \}$. 

$$<u, v> \text{ Edges iff } O_u \Delta O_v .$$

The SIG, $G = (V, E)$ has at most $2n$ vertices and $O(n)$ edges. Acyclicity can be tested in time $O(|V|+|E|) = O(n)$ time.

For two character taxa with arbitrary number of states the perfect phylogeny problem has an efficient solution.

**Rate of Evolutionary Changes**

Taxa of nucleotide or amino acid sequences.

Given two taxa $s_i$ and $s_j$, measure their distance

$\text{Distance}(s_i, s_j), d_{ij} =$ Edit distance based on pairwise sequence alignment.

Assumptions about the Molecular Clock (governing rate of evolutionary change):

- Only independent substitutions
- No back or parallel mutations
- Neglect selection pressure.

**Amino Acid Sequences**

- Amino Acid substitution rate per site per year. varies between organisms and protein classes

Example:

- for guinea pig insulin $\frac{1}{4} 5.3 \times 10^{-9}$
- for other organisms $\frac{1}{4} 0.33 \times 10^{-9}$

Other Examples of:

- Fibrinopeptide $\frac{1}{4} 9 \times 10^{-9}$
- Histone $\frac{1}{4} 1 \times 10^{-11}$

**Estimating**

$X$ & $Y =$ homologous proteins of same length $n$

$n_d =$ Number of differences between homologous amino acid sites.

$X$ and $Y$ are isolated from two distantly related species that diverged $t$ time ago.

$p = \frac{1}{4} \frac{n_d}{n} =$ Probability of an amino acid substitution occurring at a given site of either $X$ or $Y$.

**Estimating (Contd.)**

$q = 1 - p = 1 - \frac{n_d}{n} =$ Pr[# mutations at site $X_1 = 0$]

$\leq \text{Pr[# mutations at site } Y_1 =0]$

$Z =$ Random variable counting the number of mutations over time $t$

at a fixed site for an amino acid sequence with substitution rate
per site per year \( \rightarrow \) Poisson(\( t \))

\[ Pr[Z = k] = \exp\left(-\frac{t}{q}\right) \left(\frac{t}{q}\right)^k / k! \]

\[ q = \exp(2t) \]

\[ = \ln(1/q)/2t. \]

**Example: Histone H4**

X & Y = Histones from cow and pea.

\( n = 105, \; n_d = 2, \; q = 1 - n_d/n = 103/105 \)

\( t = 10^9; \) Plants and animals diverged about a billion years ago.

\[ = (1/2t) (-\ln (1 - n_d/n)) \]

\[ \frac{1}{4} \left(\frac{n_d/n}{2t}\right) \]

\[ \frac{1}{4}(2 \times 10^{-2})(2 \times 10^9) \times 10^{-11} \]

**Other Approaches**

BLOSUM matrix

PAM (Accepted Point Mutation) matrix

WAC (Wei-Altman-Chang) matrix

**Nucleotide Sequences**

Synonymous or Neutral Substitutions:

- Nucleotide substitutions with no effect on expressed amino acid sequences
  - Genetic code is redundant—Most substitutions to 3rd positions are synonymous.
  - Often a single non-synonymous nucleotide substitution is likely to change one amino acid into a related amino acid (e.g., both hydrophobic).

Molecular clock is modeled based on non-synonymous substitution rate.

**Variability of Nucleotide Mutation Rate**

**Transitional Mutations:**
- purine-purine, i.e. A $ G
- pyrimidine-pyrimidine, i.e. C $ T

**Transversal Mutations:**
- purine-pyrimidine: A $ T, A $ C, G $ C, G $ T

Usually transitional mutations are more likely. Mutation into A is more likely.

**Effect of DNA repair mechanism**
- for higher primate \( \frac{1}{4} \times 1.3 \times 10^{-9} \) site/yr
- for sea urchins & rodents \( \frac{1}{4} \times 6.6 \times 10^{-9} \) site/yr
- for mammalian mtDNA \( \frac{1}{4} \times 10^{-8} \) site/yr
- for plant cpDNA \( \frac{1}{4} \times 1.1 \times 10^{-9} \) site/yr
Markov Process Model of Mutation

Evolution is modeled by a stochastic process, $X(t)$ with real-valued time parameter $t = 0$

A time-homogeneous Markov process $(Q, \pi, P(t))$

$Q = \{A, C, G, T\} =$ States

$\pi = \{A, C, G, T\} =$ Initial Distribution

$P(t) = \ldots$

Markov Process (Contd.)

$p_{i,j}(t) = \Pr[X(t) = j | X(0) = i] = \begin{cases} \frac{k_2\{A,C,G,T\} \cdot p_{k,i}(t)}{8t} \end{cases}$

is a stationary distribution for $P(t)$

Markov Process (Contd.)

$P(t) = e^{-4\left(I - \frac{1}{4}\right)t}$

$= \begin{cases} \sum_{n=0}^{\infty} \left(-4t\right)^n/n! \end{cases}$
\[ = \text{I} \text{e}^{4t} \{ \text{I} + (\text{e}^{4t} - 1) \} \]
\[ = \text{e}^{-4t} \text{I} + (1 - \text{e}^{-4t}) \]
\[ p_{i,i}(t) = \frac{1}{4} (1 + 3 \text{e}^{4t}) \]
\[ p_{i,j}(t) = \frac{1}{4} (1 - 4 \text{e}^{-4t}), \quad i \neq j. \]